

IN THE CLAIMS:

Please AMEND claims 1, 7 and 14 and ADD claims 19 and 20 in accordance with the following:

1. (Currently amended) A fixed length data search device, comprising:
 - a hash operation means for applying first and second similarly constructed hash functions~~operating~~ and thereby outputting multiple entry data corresponding to respective a-first and second hash values of an inputted fixed length datum;
 - a data table memory consisting of N numbers of memory banks, where N is an integer greater than or equal to 2, the data table memory capable of storing a data table holding a large number of fixed length data;
 - a pointer table memory for storing ~~a~~ a main memory pointer table, which is associated with the first hash function, and a subordinate memory pointer table for use when the main memory pointer table is filled to a predetermined level with respect to the N numbers of memory banks, which is associated with the second hash function, -that each indicates a memory address in said data table memory at which each fixed length datum is stored in said data table memory with said first and second hash values each acting as an~~a~~ respective index therefore;
 - a pointer selector table to indicate which one of said main and subordinate memory pointer tables should be referred to when a fixed length datum is inputted; and
 - a comparison means for simultaneously comparing a plurality of fixed length data stored at the same memory address in said N numbers of memory banks, the comparison means for outputting results of the comparison.

2. (Original) The fixed length data search device according to claim 1, wherein said comparison means comprises N numbers of comparators for determining if two fixed length data are identical, the device referring to said memory pointer table based on a resulting memory address, said comparison means determining if any of the fixed length data stored at the same memory address in said N numbers of memory banks matches the single fixed length datum inputted to said hash operation means, said comparison means outputting the result of the determination.

3. (Original) The fixed length data search device according to claim 1, wherein an datum identical to the single fixed length datum inputted to said hash operation means is searched in said data table through said hash operation means, said single fixed length datum registered in said data table if the datum has not been previously registered with said data table.

4. (Original) The fixed length data search device according to claim 3, wherein each of a plurality of fixed length data having the same hash value are stored at the same memory address of a different memory bank in said data table memory.

5. (Original) The fixed length data search device according to claim 3, wherein each of a plurality of fixed length data having a different hash value are stored at the same memory address of a different memory bank in said data table memory.

6. (Original) The fixed length data search device according to claim 1, wherein said fixed length data is a MAC (Media Access Control) address for network communications, and said data table memory is a MAC entry table memory for storing a MAC address table holding a large number of MAC addresses.

7. (Currently amended) A fixed length data search device, comprising:
a hash operation means, said hash operation means using two types of similarly constructed hash functions to determine a first and second hash values of an inputted fixed length datum wherein said first and second hash values includes multiple entry data;

a data table memory consisting of N numbers of memory banks, where N is an integer that is greater than or equal to 2, the data table memory for storing a data table holding a large number of fixed length data;

a pointer table memory for storing a first memory pointer table, said pointer table memory that indicates a memory address in said data table memory at which each fixed length datum is stored in said data table memory wherein said first hash value is an index, and a second memory pointer table for use when the first memory pointer table is determined to be filled to a predetermined level with respect to the N numbers of memory banks, which is associated with

the second hash function, holding the memory address in said data table memory at which each fixed length datum is stored in said data table memory, said second hash value as an index; and a comparison means for simultaneously comparing a plurality of fixed length data stored at the same memory address in said N numbers of memory banks, the comparison means for outputting results of the comparison.

8. (Original) The fixed length data search device according to claim 7, further comprising a pointer selector table using said first hash value as an index to indicate which one of said first and second memory pointer tables should be referred to when a fixed length datum is inputted.

9. (Original) The fixed length data search device according to claim 8, wherein when the number of stored data of separate fixed length data having the same first hash value exceeds N, a pointer in said pointer selector table corresponding to the first hash value of an unstored fixed length datum stored is set to said second memory pointer table, said memory address at which the datum is stored managed with said second memory pointer table.

10. (Original) The fixed length data search device according to claim 9, wherein said comparison means comprises N numbers of comparators, said comparators simultaneously compare all bits to determine whether or not two fixed length data are identical.

11. (Original) The fixed length data search device according to claim 9, wherein said comparison means determines if any of the fixed length data stored at the same memory address in said N numbers of memory banks matches the single fixed length datum inputted to said hash operation means and outputs the result of the determination.

12. (Original) The fixed length data search device according to claim 9, wherein if another fixed length datum having the same first hash value as an inputted fixed length datum has not been registered with said data table, said inputted fixed length datum is stored in said data table memory, and said memory address at which the datum is stored is managed with said main memory pointer table.

13. (Original) The fixed length data search device according to claim 7, wherein said fixed length data is a MAC (Media Access Control) address for network communications, and said data table memory is a MAC entry table memory for storing a MAC address table holding a large number of MAC addresses.

14. (Currently amended) A method of searching fixed length data, comprising the steps of:

performing first and second similarly constructed hash operations ~~said hash operation to~~ thereby outputting a respective first and second hash values of inputted fixed length data, wherein each of said hash values includes multiple entry data;

referring to a main memory pointer table, which is associated with the first hash operation, or a subordinate memory pointer table for use when the main memory pointer table is filled to a predetermined level with respect to N numbers of memory banks, which is associated with the second hash function, which is associated with the second hash operations, each of which holding holds a memory address in a data table memory at which each fixed length datum is stored in said data table memory with said first and second hash values each acting as an a respective index therefor;

reading N numbers of fixed length data stored at an address pointed to by a pointer in said memory pointer table from a data table stored in said data table memory consisting of N numbers of memory banks, where N is an integer that is greater than or equal to 2, the data table capable of storing a large number of fixed length data, and

simultaneously comparing said read N numbers of fixed length data with said inputted single fixed length datum, and outputting results of the comparison.

15. (Original) The method of searching fixed length data according to claim 14, wherein said step of comparing comprises simultaneously comparing said read N numbers of fixed length data using parallel processing, said comparing determining if two fixed length data are identical.

16. (Original) The method of searching fixed length data according to claim 15, wherein said comparing comprises the steps of: searching an identical datum to said inputted single fixed length datum in said data table based on its hash value, and registering said inputted single fixed length datum in said data table if said identical datum has not been detected in said step of searching.

17. (Original) The method of searching fixed length data according to claim 16, wherein each of separate fixed length data having the same hash value is registered with the same memory address of a different memory bank in said data table memory during said registering.

18. (Original) The method of searching fixed length data according to claim 17, wherein each of a plurality of fixed length data having a different hash value is registered with the same memory address of a different memory bank in said data table memory.

19. (New) The method of searching fixed length data according to claim 16, further comprising:

detecting an exist bit associated with said data table into which the inputted single fixed length datum is to be registered;

determining if the exist bit indicates an on-state or an off-state of said data table; and

when the exist bit indicates the on-state, proceeding with the registering and, when the exist bit indicates the off-state, proceeding with the second hash operation.

20. (New) The method of searching fixed length data according to claim 19, where the exist bit indicates the off-state, further comprising indicating that a next registered single fixed length datum is to be registered in the subordinate memory pointer table.